

SUPERVISOR HISTORY SERVICE IMPORT MANAGER

BACKGROUND

[0001] The present disclosure pertains to controllers and particularly history service of controllers.

SUMMARY

[0002] The disclosure reveals a system and approach of importing data from site controllers as history imports with a history service by a supervisor in conjunction with a host processor. The history service may reconfigure the history imports to provide the history service virtually full control over the history imports. The history service may run a history import cycle to ensure that virtually all enabled history imports are performed. The history service may establish a list of history imports to be processed at a beginning of the history import cycle. The history import cycle may query virtually all history imports in an order on a last success time of a history import operation where an oldest time is first on the list. The history service may ensure a maximum number of history imports are running by querying for a count of history imports of an in progress state.

BRIEF DESCRIPTION OF THE DRAWING

[0003] FIG. 1 is a diagram of a history import management arrangement for a controller.

DESCRIPTION

[0004] The present system and approach may incorporate one or more processors, computers, controllers, user interfaces, wireless and/or wire connections, and/or the like, in an implementation described and/or shown herein.

[0005] This description may provide one or more illustrative and specific examples or ways of implementing the present system and approach. There may be numerous other examples or ways of implementing the system and approach.

[0006] A Novar™ Opus™ supervisor may be a target application to provide a present feature. A user of the Novar Opus supervisor may manage and communicate with hundreds or thousands of remote site controllers from a centralized location via an intranet. A site controller may be referred to as an XCM controller or XCM. The XCM in turn may manage and communicate with tens or hundreds of field controllers within the site that perform real-time control of building equipment such as HVAC units, lighting panels, metering and refrigeration circuits.

[0007] An existing Opus supervisor may have an ability to pull history data from points being logged with remote site XCM controllers. An approach to perform the pull history data task may be to create within the Opus supervisor application a history import object for each point or group of points. Due to a scale of a customer environment, history data may be pulled from thousands to millions of history import objects. A design of the history import object may be as follows. The object may function as an independent execution unit that functions without regard to its impact on the operating environment, the single Opus supervisor application (process).

[0008] The history import object may be configured with a trigger time of day setting which when the time is reached,

will initiate the object's action to connect to the remote XCM and begin to transfer new history data from the XCM and write to an Opus SQL server database. This may take several minutes to complete, depending on the host server size and other asynchronous operations within the Opus supervisor process. Typically, virtually all the history import objects trigger times may be configured to trigger randomly over a 24 hour period so that one would not intentionally trigger too many trigger times at the same time. Also, there may be a fault retry timer so that if the point encounters a fault condition, it may be rescheduled to re-trigger the history import again some number of minutes later. An issue that Novar may encounter with this design is that runtime conditions vary, and due to slowing performance of the operating environment, history objects would also slow down, causing more to trigger asynchronously compounding a slow performance condition. Then some history imports may fault resulting in the history import fault retry timer to retriggering the history import again. The result may be that there are so many simultaneous history imports running that the single Opus supervisor process could become un-usable even to the point that it would no longer be communicated with. Also, another compounding factor is that as these failure conditions occur, an importation of new history data may get farther behind causing history imports to take longer to complete contributing to more imports running, simultaneously. Through diagnostic efforts, it appeared to be found that an Opus supervisor process was consuming nearly 100 percent CPU (central processing unit) and large amounts of the host server system memory. The only recourse appeared to kill the Opus supervisor and then restart the supervisor.

[0009] The present approach may provide a solution to the stated issue by providing the following product features and the necessary user activities required to use these features. The Opus supervisor may provide a new Opus history service with improved runtime management of the history import operations. A new service operation may be the present feature.

[0010] An operational goal of the new Opus history service may be to intelligently manage and control the multiple history import processes ensuring that the history data is collected in a timely manner and at the same time not to allow the imports to overtake host server resources thus causing the Opus supervisor to become unstable or unusable.

[0011] The Opus history service will not necessarily replace the history import objects described as central to the issue to resolve. Instead, the service may re-configure the imports so they can be centrally managed by the history service. A first operational action of the service may be to reconfigure the history import objects by disabling the existing invocation timer and retry timer that cause a failure condition. Once the timers are disabled, the service may have full control over the history import objects.

[0012] The Opus history service may be designed to run a scheduled history import cycle to ensure that virtually all enabled history imports are performed. Within the cycle, there may be a periodic interval timer allowing the host operational conditions to be monitored, and for the management of the history import objects execution. At the beginning of an import cycle, the service may establish a complete list of history imports that will be processed. The service cycle may initially query for all the history import objects